**Gáy**

LÊ THỊ HỒNG QUYÊN MSSV: N14CCCN059 Tên Đề tài: **Xây dựng ứng dụng Chat trên Android** Lớp: C14CQCP01-N 2016 - 2017

**TẬP ĐOÀN BƯU CHÍNH VIỄN THÔNG VIỆT NAM**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

------------------------------



**BÁO CÁO ĐỊNH KỲ**

**THỰC TẬP TỐT NGHIỆP**

***Đề tài*: “** **Xây dựng ứng dụng Chat trên Android”**

**Người hướng dẫn : HỒ ĐẮC HƯNG**

**Sinh viên thực hiện : LÊ THỊ HỒNG QUYÊN**

**Mã số sinh viên : N14CCCN059**

**Lớp : C14CQCP01-N**

**Khoá** **: 2014-2017**

**Hệ** **: CAO ĐẲNG CHÍNH QUY**

**TP.HCM, tháng 3 /2017**

TP. HCM

2017

**TẬP ĐOÀN BƯU CHÍNH VIỄN THÔNG VIỆT NAM**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

------------------------------



**BÁO CÁO ĐỊNH KỲ**

**THỰC TẬP TỐT NGHIỆP**

***Đề tài*: “ Xây dựng ứng dụng Chat trên Android”**

**Người hướng dẫn : HỒ ĐẮC HƯNG**

**Sinh viên thực hiện : LÊ THỊ HỒNG QUYÊN**

**Mã số sinh viên : N14CCCN059**

**Lớp : C14CQCP01-N**

**Khoá** **: 2014-2017**

**Hệ** **: CAO ĐẲNG CHÍNH QUY**

**TP.HCM, tháng 3 /2017**

**MỤC LỤC**

**Mở đầu**:

Trang

**CHƯƠNG 1: Web API**

**4**

* 1. **Giới thiệu về API**
  2. **Giới thiệu Web API**

**CHƯƠNG 2: Lập trình Android**

**7**

**2.1. Tổng quan về hệ điều hành Android**

**2.2. Giao diện**

**2.3. Ưu và nhược điểm**

**CHƯƠNG 3: Phân tích, thiết kế cơ sở dữ liệu**

50

**KẾT LUẬN 120**

**PHỤ LỤC 121**

***Ghi chú:***

* In đậm và in hoa tiêu đề của các chương, mục lớn (toàn bộ size: 12)
* Chữ số thứ nhất chỉ thứ tự chương
* Chữ số thứ 2 chỉ thứ tự mục trong chương
* Chữ số thứ 3,…,chỉ thứ tự các tiểu mục

**CHƯƠNG 1: Web API**

**1.1 Giới thiệu về API**

- Application Programming Interface: Giao diện ứng dụng lập trình

- API là các hàm , phương thức để cho các ứng dụng bên ngoài có thể gọi , tương tác để trao đổi thông tin , tính toán.

- Việc trao đổi này giúp các nhà lập trình tạo ra các service  hỗ trợ những lập trình viên khác có thể tương tác với ứng dụng của chính mình.

- Hiện nay trên web các dịch vụ của google, facebook cung cấp rất nhiều api để lập trình viên có thể xây dựng tương tác giữa website của họ với google, facebook.

- Mỗi phần mềm, ứng dụng có các cung cấp các API để các ứng dụng khác có thể tương tác với nó. Và việc xây dựng lên các API để cho các ứng dụng bên ngoài cũng cần tuân thủ các chuẩn công nghệ để nhiều nền tảng công nghệ có thể sử dụng được API mà ứng dụng cung cấp.

**1.2 Giới thiệu web api**

- Web API là một phiên bản kế tiếp của WCF, giúp chúng ta xây dựng lên các Service cung cấp dịch vụ cho các ứng dụng web, window…

- Trước web API chúng ta để có các service API chúng ta phải cấu hình, xây dựng các ứng dụng wcf, web service khá phức tạp.

- Một ứng dụng đơn giản như là: Chúng ta có 1 Web API cung cấp các dịch vụ lưu trữ dữu liệu, cung cấp các chỉ số chứng khoán, kết quả bóng đá, xổ số…

- Các ứng dụng Client như website, ứng dụng winform, wpf có thể kết nối vào Web API để lấy các dữ liệu về xử lý, cũng như cập nhật thông tin lại Web API.

- Web API dùng phương thức trao đổi dữ liệu là HTTP, kiểu dữ liệu trao đổi là JSON, một chuẩn dữ liệu hướng đối tượng được dùng khá nhiều trong việc lưu chuyển thông tin trên Internet.

- Do dùng JSon là kiểu dữ liệu chuyển đổi nên tốc độ các trang web sử dụng web API tương tác dữ liệu có tốc độ khá cao. Ngoài ra do Web API dùng giao thức HTTP nên hầu như tất cả các ứng dụng trên các công nghệ đều có thể kết nối tới để lấy cũng như tương tác với web API cụ thể như chúng ta có thể dùng các công nghệ web như: Asp.net ( MVC, Web Page, Web form), PHP,jsp hay các ứng dụng desktop như: winform, wpf đều có thể dễ dàng kết nối tới web API.

- Với Web API chúng ta có thể xây dựng và phân tách các ứng dụng web lớn. Cấu hình từng thành phần riêng biệt của website. Đâu là tầng data, đâu là tầng xử lý, đâu là tầng dịch vụ… Nền tảng của các ứng dụng lớn luôn là các service để các website thành viên có thể kết nối tương tác dữ liệu.

- Do đó với Web API chúng ta có thể ứng dụng vào các dự án Web ( cũng như window) lớn để phát triển trên nhiều tầng xử lý khác nhau.

- Dùng web API chúng ta dễ dàng xây dựng các ứng dụng window kiểu điện toán ( dữ liệu ở server) còn client chỉ cài giao diện .

Hay có thể xây dựng các website Single Page Application (SPA) (tất cả web chỉ gói gọn trong 1 trang). Ứng dụng này tương tác khá cao với người dùng, tốc độ nhanh (do dùng ajax) thường được dùng làm các website tương tác với các thiết bị di động (các thiết bị di động thường có kết nối Internet chậm).

**CHƯƠNG 2: Lập trình Android**

**2.1. Tổng quan về hệ điều hành Android**

- Android là hệ điều hành mã nguồn mở, dựa trên Linux Kernel, dành cho các thiết bị di động nói chung (điện thoại, máy tính bảng, đồng hồ thông minh, máy nghe nhạc,…).

- Có nghĩa là Android không chỉ giới hạn trong phạm vi một hệ điều hành cho điện thoại! Nó có thể được nhà sản xuất cài đặt lên đồng hồ, máy nghe nhạc, thiết bị định vị GPS, thậm chí là ô tô (các thiết bị Android Auto).

- Android cũng không phải là một thiết bị hay sản phẩm cụ thể, nó là một hệ điều hành dựa trên Linux, nguồn mở, linh hoạt. Có khả năng tùy biến rất cao và có thể chạy trên nhiều thiết bị, nhiều kiến trúc vi xử lý (ARM / x86).

- Ban đầu, Android được phát triển bởi Tổng công ty Android, với sự hỗ trợ tài chính từ Google, sau này được chính Google mua lại vào năm 2005 và hệ điều hành Android đã ra mắt vào năm 2007. Chiếc điện thoại đầu tiên chạy Android là HTC Dream được bán vào ngày 22 tháng 10 năm 2008.

- Chính mã nguồn mở cùng với giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên được điều chỉnh và phân phối Android một cách tự do. Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới. Android chiếm 75% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 3 năm 2012, với tổng cộng 500 triệu thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Tháng 10 năm 2012, đã có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ Google Play (cửa hàng ứng dụng chính của Android) ước tính khoảng 25 tỷ lượt. Hiện nay con số này đã giảm xuống do sự ảnh hưởng lớn của iOS từ Apple và một phần nhỏ của Windows Phone, tuy nhiên Android vẫn dẫn đầu thị phần.

- Tính đến nay, Android đã có các phiên bản (kèm tên mã) lần lượt là:

* Android 1.5 Cupcake
* Android 1.6 Donut
* Android 2.1 Eclair
* Android 2.2 Froyo
* Android 2.3 Gingerbread
* Android 3.2 Honeycomb – Phiên bản Android đầu tiên được thiết kế cho máy tính bảng.
* Android 4.0 Ice Cream Sandwich
* Android 4.1 Jelly Bean
* Android 4.2 Jelly Bean
* Android 4.3 Jelly Bean
* Android 4.4 KitKat
* Android 5.0 Lollipop
* Android 6.0 Marshmallow
* Android 7.0 Nougat

- Về kiến trúc của hệ điều hành Android, chúng ta có thể xem qua sơ đồ sau:



- Các tầng của kiến trúc này như sau (từ trên xuống):

* Tầng Applications: Là tầng chứa các ứng dụng Danh bạ, Gọi điện, Trình duyệt, Nghe nhạc,… các ứng dụng này thường mua máy về chúng ta đã có sẵn rồi.
* Tầng Framework: Là tầng chứa các API để làm việc với hệ điều hành như lấy thông tin danh bạ, quản lý các Activity (Activity là gì thì giờ chúng ta chưa cần quan tâm, các bài sau sẽ giải thích kĩ), quản lý địa điểm, quản lý các View (cũng chưa cần quan tâm).
* Tầng Libraries: Chứa các thư viện, API gần như là cốt lõi của Android, bao gồm bộ quản lý bề mặt cảm ứng (Surface Manager), OpenGL (phục vụ cho việc dựng đồ họa phức tạp),…
* Tầng Android Runtime: Chứa các thư viện lõi của Android và máy ảo Dalvik Virtual Machine (từ Android 4 trở lên chúng ta có thêm máy ảo ART).
* Tầng Kernel: Là nhân lõi của hệ điều hành, chứa các tập lệnh, driver giao tiếp giữa phần cứng và phần mềm của Android.

- Trong quá trình làm việc, chúng ta sẽ gần như chỉ làm việc với tầng xanh lam (Applications và Application Framework) và xanh lá (Libraries). Chương trình Android được viết bằng ngôn ngữ Java và được máy ảo DVM / ART trong mỗi thiết bị Android biên dịch ra mã máy.

**2.2. Giao diện**

**-** Giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp, sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt, chạm, kéo dãn và thu lại để xử lý các đối tượng trên màn hình.

****

Giao diện Android 5.0 trên điện thoại

- Các thiết bị Android sau khi khởi động sẽ hiển thị màn hình chính, điểm khởi đầu với các thông tin chính trên thiết bị, tương tự như khái niệm desktop trên máy tính để bàn. Màn hính chính Android thường gồm nhiều biểu tượng (icon) và tiện ích (widget). Giao diện màn hình chính của Android có thể tùy chỉnh ở mức cao, cho phép người dùng tự do sắp đặt hình dáng cũng như hành vi của thiết bị theo sở thích.

- Những ứng dụng do các hãng thứ ba có trên Google Play và các kho ứng dụng khác còn cho phép người dùng thay đổi "chủ đề" của màn hình chính, thậm chí bắt chước hình dáng của hệ điều hành khác như Windows Phone hay iOS. Phần lớn những nhà sản xuất, và một số nhà mạng, thực hiện thay đổi hình dáng và hành vi của các thiết bị Android của họ để phân biệt với các hãng cạnh tranh.

**2.3. Ưu và nhược điểm**

- Ưu điểm:

+ Là hệ điều hành có mã nguồn mở nên khả năng tuỳ biến cao, có thể tùy ý chỉnh sửa mà không có sự can thiệp hay cấm cản từ Google.

+ Đa dạng sản phẩm, rất nhiều hãng điện thoại, thiết bị công nghệ đã ưu ái chọn Android cho thiết bị của họ, giá cả thì hợp lý từ bình dân đến cao cấp.

+ Kho ứng dụng Google Play Store đồ sộ.

+ Thân thiện và dễ sử dụng.

+ Khả năng đa nhiệm, chạy cùng lúc nhiều ứng dụng cao.

- Nhược điểm:

+ Dễ nhiễm phần mềm độc hại và virus. Do tính chất mã nguồn mở, nhiều phần mềm không được kiểm soát có chất lượng không tốt hoặc lỗi bảo mật vẫn được sử dụng.

+ Kho ứng dụng quá nhiều dẫn đến khó kiểm soát chất lượng, thiếu các ứng dụng thật sự tốt.

+ Sự phân mảnh lớn. Trong khi một số thiết bị Android xuất sắc đã trình làng như Galaxy S7, Galaxy Note 7, Xperia XZ…, vẫn còn rất nhiều sản phẩm giá rẻ bình thường khác.

+ Cập nhật không tự động với tất cả thiết bị. Khi một phiên bản hệ điều hành mới ra mắt, không phải tất cả sản phẩm đều được cập nhật, thậm chí nếu muốn trải nghiệm bạn thường xuyên phải mua mới thiết bị.

**CHƯƠNG 3: Phân tích, thiết kế cơ sở dữ liệu**

Cơ sở dữ liệu: pgAdmin4

CREATE TABLE User(

id serial NOT NULL,

username text NOT NULL UNIQUE,

password text NOT NULL,

firstname text,

lastname text,

age integer,

gender text,

image text,

status text,

PRIMARY KEY (id)

);